

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»  
Кафедра комп'ютерних інтелектуальних систем та мереж

МЕТОДИЧНІ ВКАЗІВКИ  
до виконання лабораторних робіт  
з дисципліни  
«ОСНОВИ АЛГОРИТМІВ»  
для здобувачів дистанційної та заочної освіти

Одеса 2025

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»  
Кафедра комп'ютерних інтелектуальних систем та мереж

МЕТОДИЧНІ ВКАЗІВКИ  
до виконання лабораторних робіт  
з дисципліни  
«ОСНОВИ АЛГОРИТМІВ»  
для здобувачів дистанційної та заочної освіти

**Затверджено**  
на засіданні кафедри комп'ютерних  
інтелектуальних систем та мереж  
Протокол № 9 від 20.02 2025.

Методичні вказівки до виконання лабораторних робіт з дисципліни «Основи алгоритмів» для здобувачів дистанційної та заочної освіти спеціальності 123 – «Комп’ютерна інженерія», освітня програма «Комп’ютерні системи та мережі» / Укл.: Є.В.Шендрик, О.В. Головачова, А.А. Нагорна – Одеса: Національний університет «Одеська політехніка», 2025. – 10 с.

Укладач:

Є.В. Шендрик, к.т.н., доцент

О.В. Головачова, ст. викл

А.А. Нагорна, асистент

## ЗМІСТ

Вступ4

|                                                         |   |
|---------------------------------------------------------|---|
| 1 Теоретична частина                                    | 6 |
| 1.1 <i>Має бути перше питання та відповідь на нього</i> |   |
| 1.2 <i>Має бути друге питання та відповідь на нього</i> |   |
| 1.3 <i>Має бути третє питання та відповідь на нього</i> |   |
| 2 Практична частина                                     | 7 |
| 2.1 <i>Має бути перше завдання та його реалізація</i>   |   |
| 2.2 <i>Має бути друге завдання та його реалізація</i>   |   |
| Список використаних джерел                              | 9 |

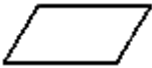
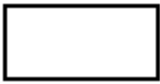
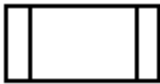
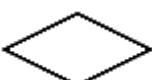
## ВСТУП

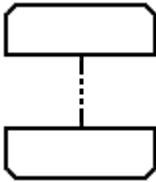
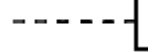
Усі лабораторні роботи складаються з розробки схеми алгоритму та відтворенні її у середовищі Microsoft Visio. При цьому робота вважається успішно виконаною, якщо, розроблена схема алгоритму, по-перше, має правильну логічну структуру, що відповідає завданню. По-друге, виконана у відповідності до ГОСТ 19.701-90 (ИСО 5807-85) Единая система программной документации (ЕСПД). Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения. По-третє, не повинна містити помилок при перевірці у середовищі Microsoft Visio.

При виконанні схем у відповідності до ЕСПД необхідно враховувати, що символи можуть бути накреслені у будь-якій орієнтації, але кращою є горизонтальна орієнтація. Всередину символу поміщують позначення або опис операцій. Напрямки ліній зв'язку зліва направо і зверху вниз вважаються стандартними, і лінії зв'язку зображуються без стрілок, в протилежному випадку - зі стрілками. У операційні блоки (процес, введення-виведення і підпрограма) входить і з них виходить тільки одна лінія зв'язку. У блок перевірки умов (розгалуження) входить одна, а виходить не менше двох ліній, близько яких записуються результати перевірки умов. З початкової вершини виходить одна лінія зв'язку, в кінцеву вершину також входить одна лінія зв'язку. Лінії можуть з'єднуватися одна з одною, але не можуть розгалужуватися. Символи схем алгоритмів можуть бути відзначені ідентифікаторами або порядковими номерами. Ідентифікатор являє собою букву або букву з цифрою і повинен розташовуватися зліва над символом. У середині з'єднувачів ставляться номери (координати) блоків до яких або від яких йдуть лінії зв'язку. Замість номера (координат) може бути поставлений деякий (один і той же в обох з'єднаннях) ідентифікатор

Таблиця 1

Умовні графічні позначення, що використовуються у схемах алгоритмів

| Найменування           | Позначення                                                                            |
|------------------------|---------------------------------------------------------------------------------------|
| Данні                  |  |
| Процес                 |  |
| Передвизначений процес |  |
| Підготовка             |  |
| Рішення                |  |

|                 |                                                                                     |
|-----------------|-------------------------------------------------------------------------------------|
| Межа циклу      |  |
| З'єднувач       |  |
| Термінатор      |  |
| Лінія           |  |
| Паралельні дії  |  |
| Пунктирна лінія |  |
| Коментар        |  |

Лабораторна робота складається з двох частин:

- теоретичної;
- практичної.

Варіанти роботи обираються згідно з наведеної нижче таблиці де:

$$\text{№\_варіанту} = \text{№\_по\_списку}.$$

Якщо у академічній групі здобувачів більше ніж 20, то обчислення наступних варіантів буде починатись з 1. Тобто 21 за списком здобувач отримає 1-й варіант, 22-й відповідно 2-й і т.д.

Таблиця 2

**Варіанти завдань**

| №<br>вар  | №№<br>теоретичних<br>запитань |    |    | №№<br>задач |    | №<br>вар  | №№<br>теоретичних<br>запитань |    |    | №№<br>задач |    |
|-----------|-------------------------------|----|----|-------------|----|-----------|-------------------------------|----|----|-------------|----|
| <b>1</b>  | 1                             | 10 | 20 | 1           | 20 | <b>11</b> | 10                            | 11 | 20 | 1           | 11 |
| <b>2</b>  | 2                             | 11 | 19 | 2           | 19 | <b>12</b> | 9                             | 12 | 19 | 2           | 12 |
| <b>3</b>  | 3                             | 12 | 18 | 3           | 18 | <b>13</b> | 8                             | 13 | 18 | 3           | 13 |
| <b>4</b>  | 4                             | 13 | 17 | 4           | 17 | <b>14</b> | 7                             | 14 | 17 | 4           | 14 |
| <b>5</b>  | 5                             | 14 | 16 | 5           | 16 | <b>15</b> | 6                             | 15 | 16 | 5           | 15 |
| <b>6</b>  | 6                             | 15 | 20 | 6           | 15 | <b>16</b> | 5                             | 16 | 20 | 6           | 16 |
| <b>7</b>  | 7                             | 14 | 16 | 7           | 14 | <b>17</b> | 4                             | 17 | 19 | 7           | 17 |
| <b>8</b>  | 8                             | 13 | 17 | 8           | 13 | <b>18</b> | 3                             | 15 | 18 | 8           | 18 |
| <b>9</b>  | 9                             | 12 | 18 | 9           | 12 | <b>19</b> | 2                             | 13 | 17 | 9           | 19 |
| <b>10</b> | 10                            | 11 | 19 | 10          | 11 | <b>20</b> | 1                             | 11 | 16 | 10          | 20 |

## 1. ТЕОРЕТИЧНА ЧАСТИНА

1. Основні етапи рішення задачі на ЕОМ.
2. У чому відмінність оператора **if(...){...}** від операторів **while(...){...}**, **do{...}while(...)**?
3. Побудуйте фрагменти СА для операторів **while(...){...}**, **do{...}while(...)**, **if(...){...}**, **if(...){...}else{...}**.
4. У яких випадках використовується оператор **switch**? У чому його відмінність від оператора **if**? Наведіть приклад його використання.
5. Які типи циклів Ви знаєте? Опишіть відмінність між ними і побудуйте схему алгоритму (СА) кожного з них.
6. Що таке масив? Якими властивостями його можна охарактеризувати?
7. Перерахуйте основні ознаки алгоритму, а так само способи його подання.
8. Побудуйте фрагменти основних блоків СА. Опишіть призначення кожного з них.
9. Які алгоритми називають логічними? Наведіть приклад логічного алгоритму.
10. Що називається **розгалуженням**. Для чого воно потрібне?
11. Структурна алгоритмізація (основні вимоги та принципи).
12. Які блоки СА називають елементарними? Що називається деталізацією СА?
13. Основні правила визначення О-якої складності алгоритму.
14. Які алгоритми називають алгоритмами з поверненням? У чому їх особливість?
15. У чому полягає принцип "Розділяй і володарюй". Наведіть приклад його використання.
16. Основні недоліки оцінки О-якої складності програм.
17. О-складність алгоритмів.
18. Наведіть структурну схему машини Тюрінга.
19. Що називається функціональною схемою машини Тюрінга?
20. Які дії виконує машина Тюрінга працює згідно зі схемою:

|                |                   |                   |
|----------------|-------------------|-------------------|
| A              | 0                 | 1                 |
| q <sub>1</sub> | q <sub>1</sub> 1R | q <sub>0</sub> 1S |

## 2. ПРАКТИЧНА ЧАСТИНА

1. Заданий одномірний масив з різними числами, що повторюються. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але без повторів, зберігаючи послідовність. Розрахувати складність алгоритму.
2. Заданий одномірний масив з різними числами. Побудувати схему алгоритму (СА), який повертає масив тільки парних чисел (зберегти послідовність). Розрахувати складність алгоритму.
3. Заданий одномірний масив з різними числами, що іноді повторюються. Побудувати схему алгоритму (СА), який повертає масив тільки з чисел, що повторювалися (зберегти послідовність). Розрахувати складність алгоритму.
4. Заданий одномірний масив з різними числами. Побудувати схему алгоритму (СА), який повертає масив у тій же послідовності, але з нулями зсунутими у початок. Розрахувати складність алгоритму.
5. Заданий одномірний масив з  $n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але циклічно зсунутими вліво на  $k$  елементів ( $k < n$ ). Розрахувати складність алгоритму.
6. Заданий одномірний масив з  $n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але циклічно зсунутими вправо на  $k$  елементів ( $k < n$ ). Розрахувати складність алгоритму.
7. Заданий одномірний масив з  $n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але відсортованих по зростанню. Розрахувати складність алгоритму.
8. Заданий одномірний масив з  $n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але відсортованих по убиванню. Розрахувати складність алгоритму.
9. Заданий двомірний масив з  $n \times n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але циклічно зсунутими вліво на  $k$  елементів ( $k < n$ ). Розрахувати складність алгоритму.
10. Заданий двомірний масив з  $n \times n$  числами. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але переставлених дзеркально відносно головної діагоналі. Розрахувати складність алгоритму.

11. Заданий двовірний масив. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але рядки якого переставлені у зворотному напрямі ( $0 \leftrightarrow N-1, 1 \leftrightarrow N-2 \dots$ ). Розрахувати складність алгоритму.
12. Заданий двовірний масив. Побудувати схему алгоритму (СА), який повертає масив тих самих чисел, але стовпці якого переставлені у зворотному напрямі ( $0 \leftrightarrow N-1, 1 \leftrightarrow N-2 \dots$ ). Розрахувати складність алгоритму.
13. Заданий двовірний масив. Побудувати схему алгоритму (СА), який повертає кількість елементів, які більше середнього значення всіх елементів масиву. Розрахувати складність алгоритму.
14. Заданий двовірний масив. Побудувати схему алгоритму (СА), який повертає кількість елементів, які менше середнього значення всіх елементів масиву. Розрахувати складність алгоритму.
15. Побудуйте схему алгоритму та визначте складність алгоритму знаходження факторіала (відповідь пояснити). Розрахувати складність алгоритму.
16. Побудуйте схему алгоритму та визначте складність алгоритму обчислення суми елементів матриці розмірністю  $m \times n$  ( $m < n$ ) (відповідь пояснити). Розрахувати складність алгоритму.
17. Побудуйте схему алгоритму та визначте складність алгоритму обчислення значень функції  $y = \sin(x)$  для  $n$  аргументів (відповідь пояснити). Розрахувати складність алгоритму.
17. Побудуйте схему алгоритму та визначте складність алгоритму обчислення значень функції  $y = \cos(x)^2$  для  $n$  аргументів (відповідь пояснити). Розрахувати складність алгоритму.
19. Побудуйте схему алгоритму та визначте складність алгоритму знаходження ступеня  $y^n$  (відповідь пояснити). Розрахувати складність алгоритму.
20. Побудуйте схему алгоритму та визначте складність алгоритму обчислення добутку елементів тривірного масиву розмірністю  $n \times n \times n$  (відповідь пояснити). Розрахувати складність алгоритму.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Паулин О.Н. Основы теории алгоритмов: Учеб. пособие: — Одесса: Автограф, 2002. — 188 с.
2. ГОСТ 19.701-90 (ИСО 5807-85) «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения», Стандартиформ – 2010. [Электронный ресурс]. URL <http://www.gostrf.com/normadata/1/4294848/4294848992.pdf>
3. ДСТУ 3008:2015 «Інформація та документація. Звіти у сфері науки та техніки. Структура та правила оформлювання» Київ: ДП «УкрНДНЦ» - 2016. [Электронный ресурс]. URL [https://science.kname.edu.ua/images/dok/derzhstandart\\_3008\\_2015.pdf](https://science.kname.edu.ua/images/dok/derzhstandart_3008_2015.pdf)